|  |
| --- |
| https://lh6.googleusercontent.com/QcftzNtI05T0Y6fjdSh1Rr2rt8oqZ1IvnLvbn1jLJ7CCyteVir3k-xBLv4SL1wAgWJsRhmmJSR0UW-RP63_GQenE4vVWv05BRoZTsmIcBccVTnfxwmsnNMvjg599x9SqZd8E3dkd |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА** - **Российский технологический университет»**  **РТУ МИРЭА** |

Институт Информационных Технологий (ИТ)

Кафедра инструментального и прикладного программного обеспечения (ИППО)

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1** | |
| **по дисциплине** | |
| «Разработка программных приложений» | |
|  | |
| Выполнил студент группы ИКБО-02-18 | Юркевич Г.А. |
| Принял ассистент кафедры ИППО | Строганкова Н.В. |

|  |  |  |
| --- | --- | --- |
| Лабораторные работы выполнены | «\_\_»\_\_\_\_\_\_\_2020 г. |  |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_2020 г. |  |

Москва 2020

**Содержание**

[1. Цель лабораторной работы 3](#_heading=h.26in1rg)

[2. Задание 3](#_heading=h.2et92p0)

[3. Поэтапное выполнение лабораторной работы 3](#_heading=h.3dy6vkm)

[4. Результат работы 4](#_heading=h.4d34og8)

[5. Вывод 4](#_heading=h.lnxbz9)

[Текст программы 4](#_heading=h.17dp8vu)

[Список использованных источников 9](#_heading=h.35nkun2)

1. Цель лабораторной работы

Целью данной лабораторной работы является ознакомление со средой разработки программных приложений Android Studio, а также создание первого простого приложения.

**2. Общие требования**

Среда разработки по умолчанию - Android Studio. Язык программирования - Java.

По своему желанию, можно использовать Kotlin, а также XCode/Swift при наличии соответствующего оборудования.

Решения должны быть размещены в виде проекта в git репозитории (github/bitbucket).

Ссылку на git-репозиторий и отчет по выполненной работе прислать на почту: strogankova-mirea@yandex.ru.

Минимальная версия API – 19/

**2. Задание**

**2.1. Вводное задание №0**

Скачать, установить и запустить Android Studio. Создать первое приложение с одной активити. Добавить кнопку и сделать какое-либо действие по нажатию на кнопку.

**2.2. Задание №1**

Реализовать приложение, состоящее из двух активити:

1) первое активити - это splash screen. Полноэкранное без ActionBar, c одной картинкой (или анимацией по желанию) по центру. Активити показывается 2 секунды, потом запускается второе и первое «умирает». Ожидание необходимо реализовать через Thread.sleep;

2) второе активити представляет собой список из 1000000 (один миллион) элементов, где четные элементы имеют серый фон (#CCCCCC), а нечетные - белый (#FFFFFF). Каждый элемент содержит картинку и текст. Картинка может быть любая, и она не меняется для элементов (выбирайте маленькую). Текст каждого элемента - это его индекс текстом (пример «cто двенадцать»). Отсчет начинается с единицы. Шрифт большой. Приложение должно нормально обрабатывать поворот экрана. При нажатии на клавишу back во втором активити приложение должно закрываться.

**3. Поэтапное выполнение лабораторной работы**

**3.1. Поэтапное выполнение задание №0**

Помещаем элементы Button и TextView на layout.activity\_main.xml. После этого в коде активити прописывается метод прослушивания нажатия на кнопку, который изменяет текст в TextView. Тестирование показало работоспособность приложения.

**3.1. Поэтапное выполнение задание №1**

Создается сплэш скрин который зависает на экране. Зависание реализовано через Thread.sleep. После зависания первое активити реализован переход на второе активити.

Был реализован адаптер для ListView расположенном во втором активити. В коде активити arraylist заполняется миллионом элементов и передается в адаптер где обрабатывается. После обработки элементы arraylist отображаются в ListView. Тестирование показало работоспособность приложения и выполнение поставленной задачи.

**4. Результат работы**

**4.1. Результат работы задания №0**

**MainActivity:**

package com.example.lab01;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

TextView myTextView;

Button myButton;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

myTextView = (TextView) findViewById(R.id.myTextView);

myButton = (Button) findViewById(R.id.myButton);

myButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

myTextView.setText("こんにちは世界");

}

});

}

}

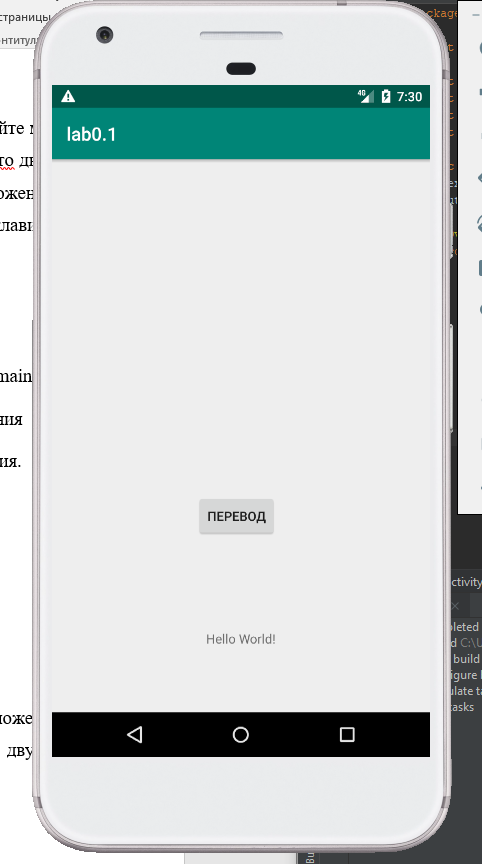


Рисунок 1 Активити перед нажатием на кнопку



Рисунок 2 Активити после нажатия на кнопку

**4.1. Результат работы задания №1**

**AndroidManifest.xml:**

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.splash">

<application

android:allowBackup="true"

android:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:roundIcon="@mipmap/ic\_launcher\_round"

android:supportsRtl="true"

android:theme="@style/AppTheme">

<activity android:name=".Secondctivity"></activity>

<activity android:name=".MainActivity" android:theme="@style/Theme.AppCompat.Light.NoActionBar">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>

**CustomArrayAdapter.java**

package com.splash;

import android.content.Context;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.ArrayAdapter;

import android.widget.LinearLayout;

import android.widget.TextView;

import java.util.ArrayList;

class Num{

public final static int DG\_POWER=6;

private final static String[][] a\_power = new String[][]{

{"0", "" , "" ,"" }, // 1

{"1", "тысяча " , "тысячи " ,"тысяч " }, // 2

{"0", "миллион " , "миллиона " ,"миллионов " }, // 3

{"0", "миллиард " , "миллиарда " ,"миллиардов " }, // 4

{"0", "триллион " , "триллиона " ,"триллионов " }, // 5

{"0", "квадриллион ", "квадриллиона ","квадриллионов "}, // 6

{"0", "квинтиллион ", "квинтиллиона ","квинтиллионов "} // 7

};

private final static String[][] digit = new String[][] {

{"" ,"" , "десять " , "" ,"" },

{"один " ,"одна " , "одиннадцать " , "десять " ,"сто " },

{"два " ,"две " , "двенадцать " , "двадцать " ,"двести " },

{"три " ,"три " , "тринадцать " , "тридцать " ,"триста " },

{"четыре ","четыре ", "четырнадцать ", "сорок " ,"четыреста "},

{"пять " ,"пять " , "пятнадцать " , "пятьдесят " ,"пятьсот " },

{"шесть " ,"шесть " , "шестнадцать " , "шестьдесят " ,"шестьсот " },

{"семь " ,"семь " , "семнадцать " , "семьдесят " ,"семьсот " },

{"восемь ","восемь ", "восемнадцать ", "восемьдесят ","восемьсот "},

{"девять ","девять ", "девятнадцать ", "девяносто " ,"девятьсот "}

};

public static String toString (int sum){

int i,mny;

StringBuffer result= new StringBuffer("");

long divisor; //делитель

int psum = sum;

int one = 1;

int four = 2;

int many = 3;

int sex = 0;

int hun = 4;

int dec = 3;

int dec2 = 2;

if(sum == 0) return "ноль ";

if(sum < 0){ result.append("минус "); psum = -psum;}

for(i=0,divisor=1; i<DG\_POWER; i++) divisor \*= 1000;

for(i=DG\_POWER-1; i>=0; i--){

divisor /= 1000;

mny = (int)(psum / divisor);

psum %= divisor;

//str="";

if(mny==0){

if(i>0) continue;

result.append( a\_power[i][one] );

}else{

if(mny>=100){ result.append( digit[mny/100][hun]); mny%=100; }

if(mny>=20 ){ result.append( digit[mny/10 ][dec]); mny%=10; }

if(mny>=10 ){

result.append( digit[mny-10 ][dec2]);

}else{

if(mny>=1 ) result.append( digit[mny][ "0".equals(a\_power[i][0]) ? 0 : 1 ] );

}

switch(mny){

case 1: result.append( a\_power[i][one] ); break;

case 2:

case 3:

case 4: result.append( a\_power[i][four]); break;

default: result.append( a\_power[i][many]); break;

};

}

}

return result.toString();

}

public static String toString(double num ){

return toString( (int)num) + "."+ toString( (int)(num\*100 - ((int)num)\*100) );

}

}

public class CustomArrayAdapter extends ArrayAdapter<String>

{

private Context context;

private ArrayList<String> stringValues;

public CustomArrayAdapter (Context context, ArrayList<String> stringValues)

{

super(context, R.layout.activity\_secondctivity, stringValues);

this.context = context;

this.stringValues = stringValues;

}

@Override

public View getView(int position, View convertView, ViewGroup parent)

{

LayoutInflater inflater = (LayoutInflater) context.getSystemService(Context.LAYOUT\_INFLATER\_SERVICE);

View view = inflater.inflate(R.layout.activity\_secondctivity, parent, false);

TextView textView = (TextView) view.findViewById(R.id.colors);

LinearLayout linearLayout = (LinearLayout) view.findViewById(R.id.llColors);

textView.setText(Num.toString(Integer.parseInt(stringValues.get(position))));

String s = stringValues.get(position);

if (Integer.parseInt(s)%2==1)

{linearLayout.setBackgroundResource(R.color.white);

}

else

{linearLayout.setBackgroundResource(R.color.gray);

}

return view;

}

}

**MainActivity.java**

package com.splash;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

//creating thread that will sleep for 10 seconds

Thread t=new Thread() {

public void run() {

try {

//sleep thread for 10 seconds, time in milliseconds

sleep(2000);

//start new activity

Intent i=new Intent(MainActivity.this,Secondctivity.class);

startActivity(i);

//destroying Splash activity

finish();

} catch (Exception e) {

e.printStackTrace();

}

}

};

//start thread

t.start();

}

}

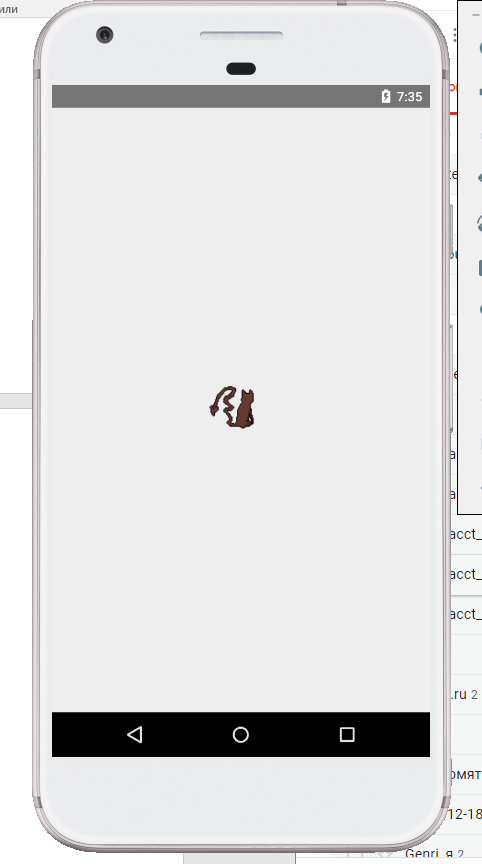


Рисунок 3 Splash screen

**Secondctivity.java**

package com.splash;

import androidx.appcompat.app.AppCompatActivity;

import com.splash.R;

import android.os.Bundle;

import android.widget.ArrayAdapter;

import android.widget.ListView;

import java.util.ArrayList;

public class Secondctivity extends AppCompatActivity {

ListView lv;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_secondctivity);

lv=(ListView)findViewById(R.id.listv);

final ArrayList<String> arrayList=new ArrayList<>();

arrayList.add(Integer.toString(1));

final CustomArrayAdapter arrayAdapter=new CustomArrayAdapter(this, arrayList);

Thread e=new Thread() {

public void run() {

try {

//sleep thread for 10 seconds, time in milliseconds

for(int i=2;i<1000001;i++){

arrayList.add(Integer.toString(i));

arrayAdapter.notifyDataSetChanged();}

finish();

} catch (Exception e) {

e.printStackTrace();

}

}

};

e.start();

lv.setAdapter(arrayAdapter);

}

@Override

public void onBackPressed() {

this.finish();

}

}



Рисунок 4 ListView

**5. Вывод**

В данной работе было скачано, установлено и запущено приложение Android Studio. Также было реализовано приложение, состоящее из двух активити. Приложение было протестировано.

**Список использованных источников**

1. Главный сайт по Android: <https://developer.android.com/>
2. Канал Google на YouTube:

<http://www.youtube.com/channel/UC_x5XG1OV2P6uZZ5FSM9Ttw>

1. Онлайн-курсы по Android:

– Udacity:

<https://eu.udacity.com/course/new-android-fundamentals--ud851>

1. <https://stackoverflow.com/>
2. <https://github.com>
3. <https://google.com/>